

REMARKS/ARGUMENTS

The specification has been amended to update the status of related applications referred to therein. No new matter has been added by any of the amendments.

Claims 1-27 are pending in the present application. Claims 1, 3, 5 and 8-27 were amended. No claims were added or canceled. Support for the claim amendments can be found, for example, on page 23, line 26 to page 24, line 4, page 29, lines 1-18, page 31, lines 3-20, and page 34, line 19 to page 35, line 2. Applicants have carefully considered the cited art and the Examiner's comments, and believe the claims patentably distinguish over the cited art in their present form. Reconsideration of the rejection is, accordingly, respectfully requested in view of the above amendments and the following comments.

I. Claim Objections

The Examiner has noted that claim 10 was not included in the original claim numbering and has required correction of the claim numbering. By the present Amendment, claims 11-28 have been renumbered as claims 10-27. In addition, the claims have been amended as necessary to properly depend from the renumbered claims. Applicants appreciate the Examiner bringing this inadvertent error to Applicants' attention.

In the following discussion, any reference to the claims relate to the claims as renumbered.

The Examiner has also objected to claims 1, 10 and 19 for misspelling of the word "automatically" in the preamble of the claims. Applicants respectfully submit that the word "autonomically" currently used in the claims is correct. The meaning of the word is "occurring involuntarily" or "automatic", and has been used throughout the specification. Withdrawal of the objection to claims 1, 10 and 19 is, accordingly, respectfully requested.

II. 35 U.S.C. § 101: Claims 10-27

The Examiner has rejected claims 10-27 under 35 U.S.C. § 101 as being directed towards non-statutory subject matter. This rejection is respectfully traversed.

In rejecting the claims, the Examiner states:

With respect to claim 10, this claim recites a data processing system for automatically patching computer program that comprises executing means, determining means, and returning means all of which can be implemented using software means only without the requirement of any type of hardware or platform to execute on. Therefore, the claim is rejected under 35 U.S.C. 101 as being Non-Statutory.

As a suggestion, Applicant could add "having processor" to the preamble of claim 11 overcome this rejection.

Regarding claim 19, this claim states, "...produced in a computer readable medium..." inside the preamble renders the claim on-statutory. Inside the specification, the Applicant defines the medium to include transmission type media such as carrier waves and signal bearing media which is considered to be Non-Statutory. Thus, the claim is rejected under 35 U.S.C. 101.

As a suggestion, Applicant could add the word "Storage" after the word "readable" inside the preamble of claim 19 to overcome this rejection.

The rejection of the base claims are incorporated into their dependent claims. Appropriate correction is required.

Office Action dated August 24, 2007, page 3.

In response, independent claims 10 and 19 have been amended as suggested by the Examiner. Claims 10-27, accordingly, now satisfy the requirements of 35 U.S.C. § 101 in all respects.

Therefore, the rejection of claims 10-27 under 35 U.S.C. § 101 has been overcome

III. 35 U.S.C. § 102, Anticipation: Claims 1-3, 10-12, and 19-21

The Examiner has rejected claims 1-3, 10-12, and 19-21 under 35 U.S.C. § 102(b) as being anticipated by Hunter, U.S. Patent No. 5,920,721 (hereinafter "Hunter"). This rejection is respectfully traversed.

In rejecting the claims, the Examiner states:

With respect to claims 1, 10, and 19, Hunter discloses a data processing system (Col 3:10-20, "...computer programs are prepared and loaded into a target computer..") for automatically patching computer program code, (Col 2:31-45, "...a loader program can patch in the better-suited code sequence..") the data processing system comprising: executing means for executing a computer program instruction, wherein the computer program instruction is located at the start of a block of code of an execution sequence; determining means for determining if metadata is associated with the computer program instruction; (Col 2:50-60, "...additional information in the executable program including an address specifying a program location..") if metadata is found, redirecting means for redirecting execution to patch instructions indicated by the metadata; (Col 2:55-60, "...a value indicating in which run-time environment the execution of one of the code sequences is better-suited..") executing means for executing the patch instructions; (Col 5:35-45, "...The resulting modified program performs the same assignment operation..") returning means for returning to an instruction of the execution sequence in the computer program. (Col 5:35-45, "...straight through to the return at location.)

Office Action dated August 24, 2007, pages 4-5.

Claim 1 as amended herein is as follows:

1. A method of autonomically patching computer program code, comprising:
 - executing a computer program instruction, wherein the computer program instruction is located at a start of a block of code of an execution sequence of original code instructions;
 - determining whether metadata is associated with the computer program instruction, wherein the metadata identifies the computer program instruction as a computer program instruction having associated patch instructions, and indicates an address of the patch instructions;
 - responsive to determining that metadata is associated with the computer program instruction, redirecting execution to the patch instructions at the address indicated by the metadata;
 - executing the patch instructions;
 - returning to an instruction of the execution sequence of original code instructions in the computer program; and
 - storing a result of executing the execution sequence.

A prior art reference anticipates the claimed invention under 35 U.S.C. § 102 only if every element of a claimed invention is identically shown in that single reference, arranged as they are in the claims. *In re Bond*, 910 F.2d 831, 832, 15 U.S.P.Q.2d 1566, 1567 (Fed. Cir. 1990). All limitations of the claimed invention must be considered when determining patentability. *In re Lowry*, 32 F.3d 1579, 1582, 32 U.S.P.Q.2d 1031, 1034 (Fed. Cir. 1994). Anticipation focuses on whether a claim reads on the product or process a prior art reference discloses, not on what the reference broadly teaches. *Kalman v. Kimberly-Clark Corp.*, 713 F.2d 760, 218 U.S.P.Q. 781 (Fed. Cir. 1983). In this case each and every feature of the presently claimed invention is not identically shown in Hunter, arranged as they are in the claims, and, accordingly, Hunter does not anticipate the claims. For example, with respect to claim 1, Hunter does not disclose or suggest “determining whether metadata is associated with the computer program instruction, wherein the metadata identifies the computer program instruction as a computer program instruction having associated patch instructions, and indicates an address of the patch instructions”, or “responsive to determining that metadata is associated with the computer program instruction, redirecting execution to the patch instructions at the address indicated by the metadata.”

Hunter describes a mechanism for creating a program having a routine that is executable on different computer versions. As shown, for example, in Figure 1 of Hunter, a loader is enabled to create different executable images 18-1 and 18-2 of the routine for the different computer versions.

In rejecting the claims, the Examiner refers to column 2, lines 50-60 of Hunter, reproduced below, as disclosing determining if metadata is associated with a computer program instruction as was originally claimed.

The present invention in another aspect is a compiler that generates, along with the code sequences, additional information in the executable program including (i) an address specifying a program location in the executable program at which the function required by the portion of the

source program is to be performed; and (ii) a value indicating in which run-time environment the execution of one of the code sequences is better suited. This additional information makes it possible for fast and efficient patching of the executable program by a cooperating entity at image activation time.

The information described in this paragraph is, in effect, information that identifies a location in a program at which a particular function is to be performed, and a value that indicates which executable image would be better suited to perform the function. The information, however, does not indicate “an address of the patch instructions” as recited in claim 1.

In Hunter, when the information indicates that a second version of an executable program is needed, loader 16 creates the second version by modifying a first version of the executable program as described in column 5, lines 35-45 of Hunter. In Hunter, accordingly, patch instructions are created by the loader when needed and are not stored at any particular address. Therefore, Hunter would appear to have no use for metadata that indicates an address of patch instructions.

Because Hunter does not disclose metadata that indicates an address of the patch instructions, Hunter also does not disclose or suggest “responsive to determining that metadata is associated with the computer program instruction, redirecting execution to the patch instructions at the address indicated by the metadata.” as also recited in claim 1.

For at least the above reasons, claim 1, as amended herein, is not anticipated by Hunter and patentably distinguishes over Hunter in its present form.

Independent claims 10 and 19 have been amended in a similar manner as claim 1 and are also not anticipated by Hunter for similar reasons as described above with respect to claim 1.

Claims 2, 3, 11, 12, 20 and 21 depend from and further restrict one of the independent claims and are also not anticipated by Hunter, at least by virtue of their dependency. Furthermore, many of these claims recite additional features that are not disclosed or suggested by Hunter. For example, claims 2, 11 and 20 specify that the patch instructions are created during execution of the computer program. The Examiner asserts that this feature is disclosed in column 5, lines 35-45 of Hunter wherein it is described that the loader modifies the executable program executed on Version 2 Alpha. Applicants respectfully disagree. Hunter discloses only that the loader modifies the executable program when it is to be executed. In fact, and as illustrated in Figure 1 of Hunter, loader 16 creates loaded executable images which are later executed. Hunter, accordingly, does not disclose that patch instructions are created during execution of the computer program, and claims 2, 11 and 20 patentably distinguish over Hunter in their own right as well as by virtue of their dependency.

Therefore, the rejection of claims 1-3, 10-12, and 19-21 under 35 U.S.C. § 102(b) has been overcome.

IV. 35 U.S.C. § 103, Obviousness: Claims 4-9, 13-18, and 22-27

The Examiner has rejected claims 4-9, 13-18, and 22-27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Hunter in view of Aoshima (U.S. Pat 5,339,426), (hereinafter Aoshima) This rejection is respectfully traversed.

In rejecting the claims, the Examiner states:

With respect to claim 4, 13, and 22, the rejection of claims 1, 10, and 19 are incorporated respectively and further,

Hunter does not disclose that the metadata is in a form of a memory word.

Aoshima discloses that the metadata is in a form of a memory word(Col 20:25-35, ". . .status word memory location.. .") in an analogous system for the purpose of providing a resume processing function which restores data indicative of the operating system operating conditions after an existing resume processing routine has been executed. (Aoshima: Col. 2: 44-47)

Therefore, it would have been obvious to a person of ordinary skill in the art at the time of the invention to disclose metadata in a form of a memory word.

The modification would have been obvious because one of ordinary skill in the art would have been motivated to provide a resume processing function which restores data indicative of the operating system operating conditions after an existing resume processing routine has been executed.(Aoshima: Col 2:44-47)

With respect to claim 5, 14, and 23, the rejection of claims 1, 10, and 19 are incorporated respectively and further,

Hunter does not disclose that the metadata includes a pointer to the patch instructions.

Aoshima discloses that metadata includes a pointer to the patch instructions. (Col 15:29-35, ". . .the virtual pointer to the real mode vector table.. .") in an analogous system for the purpose of providing a resume processing function which restores data indicative of the operating system operating conditions after an existing resume processing routine has been executed.(Aoshima: Col 2:44-47)

Therefore, it would have been obvious to a person of ordinary skill in the art at the time of the invention to disclose metadata including a pointer to the patch instructions.

The modification would have been obvious because one of ordinary skill in the art would have been motivated to provide a resume processing function which restores data indicative of the operating system operating conditions after an existing resume processing routine has been executed.(Aoshima: Col :44-47)

With respect to claim 6, 15, and 24, the rejection of claims 5, 14, and 23 are incorporated respectively and further,

Hunter does not disclose that the pointer to the patch instructions includes a starting address of the patch instructions in an allocated memory location.

Aoshima discloses that the pointer to the patch instructions includes a starting address of the patch instructions in an allocated memory location (Col 10:45-60, "...the address of the original protected mode...") in an analogous system for the purpose of providing a resume processing function which restores data indicative of the operating system operating conditions after an existing resume processing routine has been executed. (Aoshima: Col 2:44-47)

Therefore, it would have been obvious to a person of ordinary skill in the art at the time of the invention to disclose a pointer to the patch instructions that includes a starting address of the patch instruction in allocated memory.

The modification would have been obvious because one of ordinary skill in the art would have been motivated to provide a resume processing function which restores data indicative of the operating system operating conditions after an existing resume processing routine has been executed.(Aoshima:Col2:44-47)

With respect to claim 7,16, and 25, the rejection of claims 6,15, and 24 are incorporated respectively and further,

Hunter does not disclose that the starting address includes at least one of an absolute or offset address.

Aoshima discloses that the starting address includes at least one of an absolute or offset address (Col 15:24-35, ". . .offset address.. . ") in an analogous system for the purpose of providing a resume processing function which restores data indicative of the operating system operating conditions after an existing resume processing routine has been executed.(Aoshima: Col 2:44-47)

Therefore, it would have been obvious to a person of ordinary skill in the art at the time of the invention to disclose the starting address including an offset address.

The modification would have been obvious because one of ordinary skill in the art would have been motivated to provide a resume processing function which restores data indicative of the operating system operating conditions after an existing resume processing routine has been executed.(Aoshima: Col 2:44-47)

With respect to claim 8, 17, and 26, the rejection of claims 1, 16, and 25 are incorporated respectively and further, Hunter does not disclose further comprising checking means for checking a new flag in a machine status register to determine whether code patching functionality is enabled.

Aoshima discloses checking means for checking a new flag in a machine status register to determine whether code patching functionality is enabled. (Col 20:15-30, "...offset checking the previously-saved flag indicating.. . ") in an analogous system for the purpose of providing a resume processing function which restores data indicative of the operating system operating conditions after an existing resume processing routine has been executed.(Aoshima: Col 2:44-47)

Therefore, it would have been obvious to a person of ordinary skill in the art at the time of the invention to check a new flag to determine if code patching is enabled.

The modification would have been obvious because one of ordinary skill in the art would have been motivated to provide a resume processing function which restores data indicative of the operating system operating conditions after an existing resume processing routine has been executed.(Aoshima: Col 2:44-47)

With respect to claim 9, 18, and 27, the rejection of claims 1, 10, and 19 are incorporated respectively and further, Hunter does not disclose that the patch instructions include reorganized instructions, instrumented alternative instructions, and hooks to build an instruction trace. Aoshima discloses that the patch instructions include reorganized instructions, (Col 10:35-45, "...a jump instruction... ") instrumented alternative instructions, (Col 10:34-45, "...the five NOP instructions will be patched.. . ")and hooks to build an instruction trace.(Col 15:29-40, "...hooking the real mode NMI vector.. . ")

in an analogous system for the purpose of providing a resume processing function which restores data indicative of the operating system operating

conditions after an existing resume processing routine has been executed.(Aoshima:Col2:44-47)

Therefore, it would have been obvious to a person of ordinary skill in the art at the time of the invention to check a new flag to determine if code patching is enabled.

The modification would have been obvious because one of ordinary skill in the art would have been motivated to provide a resume processing function which restores data indicative of the operating system operating conditions after an existing resume processing routine has been executed.(Aoshima: Col 2:44-47)

Office Action dated August 24, 2007, pages 6-12.

The Examiner bears the burden of establishing a *prima facie* case of obviousness based on the prior art when rejecting claims under 35 U.S.C. §103. *In re Fritch*, 972 F.2d 1260, 23 U.S.P.Q.2d 1780 (Fed. Cir. 1992). Additionally, all limitations of the claimed invention must be considered when determining patentability. *In re Lowry*, 32 F.3d 1579, 1582, 32 U.S.P.Q.2d 1031, 1034 (Fed. Cir. 1994). Therefore, no *prima facie* obviousness rejection can be established if the proposed combination does not teach all of the features of the claimed invention.

In the present case, the Examiner has failed to establish a *prima facie* case of obviousness because the references, considered alone or in combination, fail to teach or suggest all of the features of the dependent claims as amended herein.

Aoshima is directed to a mechanism for restoring a processing routine after power to a computer system has been removed and later re-established. More particularly, Aoshima is directed to adapting an existing resume processing mechanism to an OS/2 operating system. Aoshima is not related to patching computer program code and does not supply the deficiencies in the independent claims as described above. Therefore, claims 4-9, 13-18, and 22-27 patentably distinguish over Hunter in view of Aoshima, at least by virtue of their dependency.

Furthermore, many of the claims recite additional features which are not disclosed in Hunter or Aoshima. For example, claims 5, 14 and 23 recite that the metadata includes a pointer to the patch instructions for identifying the address of the patch instructions. Aoshima discloses a virtual pointer to a real mode vector table in connection with a process for hooking real mode and protected mode vectors described with reference to Figure 6 of Aoshima. This disclosure in Aoshima of a virtual pointer to a real mode vector table would not suggest modifying Hunter to include a pointer to the patch instructions to identify the address of the patch instruction, because, as pointed out in the discussion with respect to claim 1, Hunter does not disclose metadata that identifies an address of patch instructions and does not need such metadata.

Claims 5, 14 and 23, accordingly, patentably distinguish over the cited art in their own right as well as by virtue of their dependency.

Claims 6-7, 15-16 and 24-25 depend from and further restrict claims 5, 14 and 23, respectively, and describe the pointer with greater specificity. These claims also patentably distinguish over the cited art in their own right as well as by virtue of their dependency.

Claims 8, 17 and 26 recite checking a flag in a machine status register to determine whether code patching functionality is to be enabled; and responsive to determining that code patching functionality is to be enabled, for enabling the code patching functionality. The Examiner refers to Column 20, lines 15-30 of Aoshima which describes a flag for indicating whether a CPU was originally in a protected mode. A disclosure of a flag that indicates whether a CPU was originally in a protected mode would not suggest to one of ordinary skill in the art to check “a flag in a machine status register to determine whether code patching functionality is to be enabled; and responsive to determining that code patching functionality is to be enabled, for enabling the code patching functionality.” Only the present disclosure contains such disclosure and the Examiner is using hindsight based on Applicants’ own disclosure in rejecting the claims. Claims 8, 17 and 26, accordingly, also patentably distinguish over the cited art in their own right as well as by virtue of their dependency.

For at least all the above reasons, neither Hunter nor Aoshima nor their combination teaches or suggests all of the features of claims 4-9, 13-18, and 22-27, and the Examiner has failed to establish a *prima facie* case of obviousness in rejecting the claims. Claim 4-9, 13-18, and 22-27, accordingly, are not obvious over Hunter in view of Aoshima and patentably distinguish over the references in their present form.

Therefore, the rejection of claims 4-9, 13-18, and 22-27 are rejected under 35 U.S.C. 103(a) has been overcome.

V. Conclusion

For at least all the above reasons, claims 1-27 patentably distinguish over the cited art and this application is believed to be in condition for allowance. It is, accordingly, respectfully requested that the Examiner so find and issue a Notice of Allowance in due course.

The Examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the Examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

DATE: November 20, 2007

Respectfully submitted,

/Gerald H. Glanzman/

Gerald H. Glanzman

Reg. No. 25,035

Yee & Associates, P.C.

P.O. Box 802333

Dallas, TX 75380

(972) 385-8777

Attorney for Applicants